The NOvA Experience on HEP Cloud: Amazon Web Services Demonstration

P Buitrago[2], S Fuess[2], G Garzoglio[2,†], A Himmel[1,2], B Holzman[2], R Kennedy[2], H Kim[2], A Norman[1,2], P Spentzouris[2], S Timm[2], and A Tiradani[2]

[1] NOvA
[2] Fermilab

[†] Corresponding author: garzoglio@fnal.gov

1 Overview

The need for computing in the HEP community follows cycles of peaks and valleys which are driven by external events including conference dates, publication deadlines, academic and holiday schedules, as well as other factors. Due to this cyclic demand behavior, classical methods of statically provisioning computing resources at providing facilities lead potentially to both over and under provisioning of resources. The static provisioning models also are unable to adapt to rapid and often transient changes in demand that arise within the typical HEP experiment's analysis cycles. As the appetite for computing increases, these inefficiencies in provisioning levels are manifested more dramatically in cost inefficiencies. One way to address the need to maximize cost efficiency by developing and applying new models for dynamically provisioning resources within the HEP domain during only those periods when they are needed.

To address this issue, the HEP Cloud project was launched by the Fermilab Scientific Computing Division in June 2015. Its goal is to develop a facility that provides a common interface to a variety of resources, including local clusters, grids, high performance computers, and community and commercial Clouds. Initially targeted communities included CMS, NOvA, and DES, as well as other Fermilab stakeholders.

In its first phase, the project has demonstrated the use of the "elastic" provisioning model offered by commercial clouds, such as Amazon Web Services. In this model, resources are rented and provisioned automatically over the Internet upon request.

In January 2016, the project demonstrated the ability to increase the total amount of global CMS resources by 58,000 cores from 150,000 cores - a 25 percent increase in preparation for the Recontres de Moriond [1]. In March 2016, the NOvA experiment has also demonstrated resource burst capabilities with 7,300 cores, achieving a scale almost four times as large as the local allocated resources and utilizing the local Amazon Web Services (AWS) S3 storage to optimize data handling operations and costs. Nova was using the same familiar services used for local computations, such as data handling and job submission, in preparation for the Neutrino 2016 conference. The cost was contained by the use of the Amazon Spot Instance Market and, later, the Decision Engine, a HEP Cloud component that aims at minimizing cost and job interruption.

This report describes the details of the work that were performed, the motivations for each of the tests or demonstrators, and the challenges and lessons learned in the use of the Fermilab HEP Cloud Facility by the NOvA experiment.

2 The NOvA Use Case

The NOvA experiment, as part of its scientific mission to analyze neutrino oscillation data from the first two years of running, requires computing resources in excess of the typical allocation that can be provided by the core Fermilab grid infrastructure. The experiment has explored multiple avenues to expand the resource pool that it can access and has moved significant fractions of its computational loads to 1) dedicated university grid sites, 2) the Open Science Grid (OSG), 3) grant based computing allocations at HPC facilities. These facilities are able to address many of the steady state computational needs of its simulation, data reconstruction and analysis activities. In addition to the steady state computations, NOvA also has identified the need to perform time sensitive computing in preparation for major conference deadlines or to perform unexpected analysis relating to a specific systematic uncertainty that arises during a study. In these cases, NOvA has considered the model of dynamically provisioned resources from commercial Clouds, using a pay-per-use model. These facilities have the potential to provide "burst" capabilities that can match the experiments demand for overnight fits, without the long term costs (in hardware and operational load) associated with the integration and monitoring of the more traditional campus grids and federations.

The NOvA experiment has worked as a partner with SCD to access cloud resources for their production computing since 2014. At the start of the investigation, the collaboration identified a specific type of cosmic-ray Monte Carlo simulation, which is required for determining backgrounds to the oscillation analysis, as a workflow well suited for commercial cloud computing. The appropriateness of this match was based on the simulation's need for no input files, aside from standard configuration files, its ability to be run entirely from a CVMFS repository image, its modest (approximately 1GB) output file, and its ratio of CPU to I/O that causes it to be predominantly CPU bound. The initial demonstration consisted of running 1000 simultaneous provisioned Virtual Machines (VM) on FermiCloud with a mapping of one compute job per allocated VM, and 1000 simultaneously running jobs on AWS in a mapping of two compute jobs per allocated VM (where the AWS VM were sized to match the memory and processing cores required to accommodate this). In this manner a simulation run to generate a dataset of 20,000 output files was successfully executed.

The demonstrator revealed limitations in the scalability of the infrastructure and hence spurred additional research and development activities to improve on the architecture that was being used at the time. The studies resulted in the development of on-demand deployment of Squid caching servers at Amazon [2] which then permitted higher levels of concurrency to be achieved in the AWS environment.

The exploration of commercial clouds was continued in the context of the HEP Cloud project. SCD and the NOvA collaboration applied and obtained a research grant from Amazon Web Services for $30,000 to advance this research. The grant funds were used to prototype and execute the demonstration of commercial clouds described in this report. The funds in the grant were made available through a credited account with AWS. The account was enabled through a procurement process that selected DLT as the reseller of AWS services [1].

The costs were further contained by negotiated agreements between AWS and the Energy Science Network (ESNet). These agreements have enabled the Data Egress Waiver program for the Fermilab AWS accounts. Under this program scientific and educational institutions connected to the internet by research networks, such as ESNet and Internet2, receive a waiver for the costs of data movement out of the AWS environment, when the data movement constitutes 15% or less of the total monthly costs for services [3].

3 Computational Campaigns

NOvA managed three separate computational campaigns on AWS through the Fermilab HEP Cloud facility. Each campaign was systematically aimed at accomplishing three complimentary goals:

- first, each campaign was designed to explore increasing the scale of resources that were provisioned compared to the previous campaign;

- second, each campaign would increase the stability of the system at the new scale and improve operational performance in both efficiency and operational support load;
- third, produce a specific useful physics result for the 2015/2016 NOvA oscillation analysis.

The operational support and primary job scheduling and submission was managed by the OPOS team in SCD and followed procedures that had been established by the group for running large scale offline production in other environments. The OPOS group also followed the same management and reporting practices already in place for this group. The data management, movement and migration, and accounting was managed by the data management group using the SAM data handling system and its associated tools.

The three campaign conducted under this model were:

1. Campaign 1: Processing of data for the sterile neutrino oscillation analysis to perform an unbiased particle identification (PID) calculation. The data being processed were simulation samples that had already been run through the NOvA particle generation, detector simulation, and reconstruction chain. The addition of the PID information was then the last step required for the collaboration to being analysis of these simulations. The actual workflow processed a large number (188,000) of small files (average 140 KB/file) for an aggregated data volume of 26.6 TB. The PID code itself required a relatively modest amount of CPU per file (8.6 minutes average) running through the PID algorithms (total aggregated CPU time: 27,000 CPU h)[1]. As the first of the campaigns, it had the goal to demonstrate the feasibility of using AWS resources for NOvA at the scale of 1,000 slots for one day. This level of concurrency is typical for the NOvA production workflows and had been shown to work well in the FNAL computing environment. This campaign was the baseline demonstrator for the integration of NOvA with the HEP Cloud project. The system successfully ran and achieved 1,500 concurrent slots.
2. Campaign 2: Simulation of the near detector neutrino fluxes using the GENIE generator to provide a critical sample of "fluxswapped" interactions. This Monte Carlo technique utilized the standard beam flux and GENIE event generator to produce an accurate beam flux corresponding to the NuMI target propagated to the NOvA detector. The flux is "swapped" through a process that reassigns neutrino flavor (mapping ν_μ events into ν_e) prior to determining the interaction point in the detector and propagating the resulting particles through a Geant-based detector simulation. This information is critical to being able to model the signatures and energy spectrum of the beam flux (specifically a ν_e component) prior to oscillations and is used in the extrapolation of the flux to the far detector for the signal prediction. The workflow processed 8.9 TB of data from 19,000 files[2]. This campaigne was differed from the first one mainly in the average size (0.5 GB/file) of the files

[1] SAM dataset: prod_reco_S15-05-04_fd_numi_full

[2] dataset: prod_daq_R160211prod2genie.a_nd_genie_fluxswap_none_fhc_nova_v08_full_v1

being processed. The goal of the campaign was to improve operational practices of running in the AWS environment at scale and to integrate the AWS Simple Storage Service (S3) for staging data for output and transfer/retrieval to FNAL. This staging of data was designed to greatly improve processing efficiency.

3. Campaign 3: Official (final) reconstruction of the near detector neutrino events. This sample was the baseline sample for performing extrapolation of neutrino signals to the far detector. The sample corresponds to the official GENIE-generated events in the forward horn current configuration (beam "neutrino mode", as opposed to "anti-neutrino mode") and did not contain flux reweighting corrections. This sample was essential to have to compute the final systematic uncertainties on the oscillation signal. The workflow processed 56 TB of data[3] spread over 57,000 files (average 1 GB/file) representing 114 million near detector events. The reconstruction time required for this stage of processing consumed approximately 4.5 hours per file constituting a total of 260,000 CPU h. This set of jobs produced 124 TB of output data. Goal of the campaign was to again improve and refine operational practices and to demonstrate a burst capacity for offsite I/O intensive workflows. This campaign modified significantly the workflow to introduce the registration of output files from the worker nodes on which they were produced. This additional registration and metadata declaration step shifted the burden of bookkeeping from a centralized File Transfer Service (FTS) system to a highly parallel grid environment. Moving the bookkeeping in this manner significantly reduces the latency between when an output file was created and when it could then be accessed for validation or additional processing. This then allows faster chaining of output to input style workflows (workflows that use the output of one stage as the input to the next stage) and allows for resubmission of failed jobs without accidental duplication of files (i.e. it removes a race condition related to how certain datasets are evaluated when work on a related or dependent dataset is ongoing). The other benefit to this workflow modification is that it enabled the use of standard dataset migration and replication tools from the SAM/SAM4Users tool suite to be directly applied to the AWS environment. This campaign also leveraged three separate AWS availability regions instead of the single region that was used previously. This expanded significantly the resources that were available to the processing. Because this campaign dominated the work in both scale and impact, it is the focus of this report.

4 Production Operations and Processes

The general workflow management and data management that was used throughout the demonstrators relied on a combination of the existing standard tools, used by NOvA computing and the production data processing group, and a set of extensions to those tools. These were specifically developed to address the integration of the "standard" data processing environment with the specialized environment of the HEPCloud and commercial clouds.

[3] dataset prod_artdaq_R16-02-11-prod2genie.a_nd_genie_nonswap_nogenierw_fhc_nova_v08_full_v1

In this model, the input datasets were specifically prestaged to predetermined locations on AWS S3 using a combination of the standard SAM4Users tool suite and modifications to the migration tools to allow for highly parallel staging of the data. The tool uses multiple nodes from the local grid cluster (GPGrid) to push files to S3 from dCache. With each node having a 1 Gbps network interface, the peak aggregate upload bandwidth saturated at 12 Gbps. This resulted in an average transfer/replication time of a few hours for the input datasets used in the campaigns.

Despite the fact that jobs were configured to run on three AWS regions to improve resource availability, the input dataset was transferred from Fermilab to only one of the AWS S3 regions[4] (Oregon). The jobs relied on the AWS internal network for data transfers. This incurred a data transfer fee that was cheaper than replicating the data three times.

Jobs were submitted with the jobsub tool to the FIFE batch submission system. The jobs were routed to HEP Cloud through an attribute that requested cloud resources explicitly. Each job processed multiple files one after the other (typically 5), asking SAM each time to transfer an unprocessed file from the dataset. If a job failed because it was preempted (e.g. the VM was overbid on the spot market), HEP Cloud automatically resubmitted the job, which resumed regular file processing via SAM. A file that was not fully processed because of preemption or application failures, however, was not transferred again by SAM[5]. After all jobs had finished, therefore, operations typically submitted recovery jobs to process all files in the dataset.

To simplify the recovery process, the input dataset was defined as a “draining” dataset i.e. a dataset that includes only unprocessed files. This way, each recovery resubmission would use the same dataset name, leaving the responsibility of identified unprocessed files to the SAM system. As usual, the dataset was defined as a named query in the SAM metadata catalog and followed the following pattern:

```
(defname: my_dataset_name and full_path like s3%) minus isparentof: ( full_path 's3:%' )
```

The first part of the query (first set of parentheses) selects all the files in the dataset that have a location on S3; the second part (*minus* ...) removes all the files that have already associated output stored in S3 (*isparentof:*). This method assumes that jobs can stage output to S3 and declare metadata and file location to SAM as soon as the file is successfully processed (§3).

At the end of a campaign, SAM4Users was used to transfer the files from S3 to dCache for long-term archiving.

5 Services

The production campaigns relied on several supporting services deployed at AWS and at Fermilab. The CMS report [1] provides a detailed description of the services. For reference into that document, NOvA used the following services:

[4] Despite the fact that S3 is a global service with a global namespace, AWS organizes S3 data in “buckets” that are stored at a given region

[5] The current implementation could be improved by marking these files as unprocessed when the job restarts, thus making them available immediately for reprocessing

Service	Deployed at	Notes
Squid / CVMFS	AWS	Local cache for software distribution. The service automatically scales in the number of instances depending on the load [2].
Network Configuration	AWS	Defines network access for instances running at AWS
AWS Limits	AWS	Defines usage limits for AWS service such as maximum number of VM, storage, etc. Limits were typically lower than CMS because the targeted scale was an order of magnitude lower.
Spot Market	AWS	Users bid on the excess VM capacity at AWS. This reduces resource price of several times the regular cost; however, it may be preempted if overbid
Accounting and Billing	FNAL	Resource accounting and monitoring; alarms on spending rate thresholds for intrusion detection
GlideinWMS	FNAL	Workload management for FIFE and internal to HEP Cloud

In addition, NOvA relied on the FIFE job submission service and SAM data handling, as discussed in §4.

6 Workflow Performance

NOvA executed three different campaigns on AWS through the HEP Cloud Facility (§3). Each campaign was designed to improve the capabilities of the system and refine the operational practices of production management. The third and last campaign brought the system to production quality and was used to characterize the performance of the workflow. We report on the details of the 2nd resubmission of the near detector reconstruction campaign.

The resubmission processed 80% of the dataset (47,000 files) in two days[6] as a single submission using production best practices. Best practices included: using S3 as local storage and standard tools for data staging in and out (SAM4Users); enabling immediate job recovery submission by declaring file metadata and location from the jobs; maximizing the available

[6] From March 22 at 3 pm CST to March 24 at 4 pm

resources from AWS by using all three AWS US regions, resulting in a burst capacity of 7,300 slots (almost four times the amount of resources dedicated to NOvA at Fermilab).

It should be noted that during these operations, about 500 jobs were affected by a network incident that caused the connections to the calibration database to hang for about 12 hours. While this reflects a realistic condition in production activities, some metrics were skewed and are presented below for both the second and the smaller subsequent recovery submission for comparison.

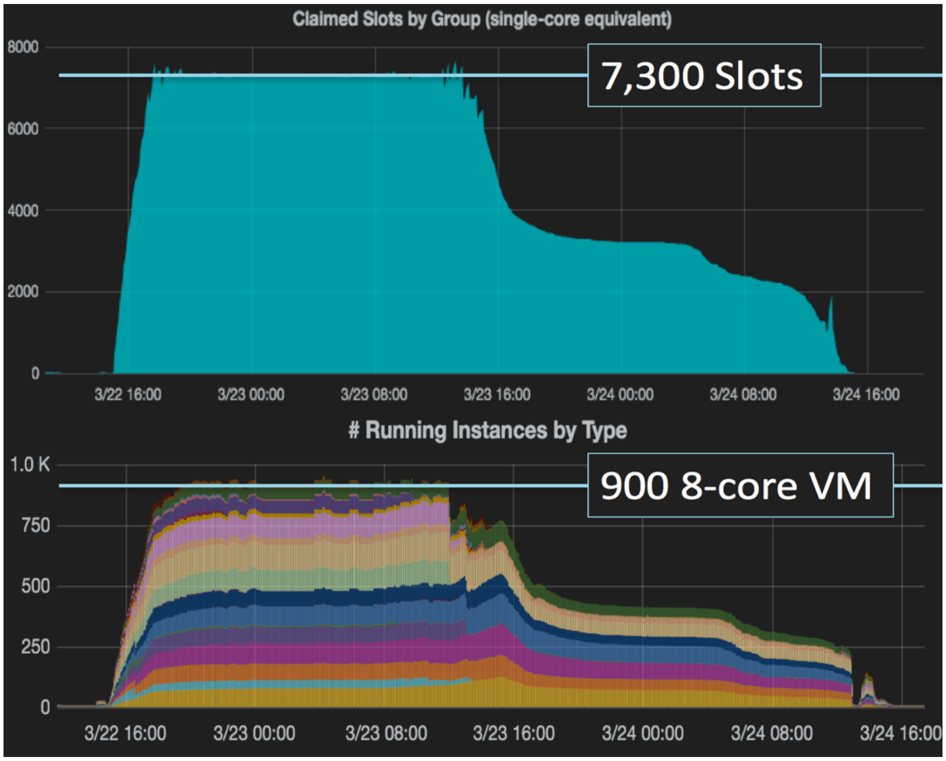


Figure 1: Total number of computing slots (top) and of virtual machines per instance types, availability zone, and region (bottom) for the third nova campaign

The HEP Cloud Facility was configured to provision 5 types of 8-core and 16-core instances[7] form AWS in 8 Availability Zones from 3 AWS Regions. This resulted in 7,300 concurrent jobs on 900 instances for 21 out of 48 hours, for a total of 203,000 core hours (Figure 1). In general, the variety of instance types is key to enabling a large scale of slots and NOvA has further opportunities to gain in diversity of resources. In particular, NOvA could not run on any of the c3 and c4 instance types, which all have less than 2 GB of memory per core, because the

[7] The instance types were m3.2xlarge, m4.2xlarge, m4.4xlarge, r3.2xlarge, and r3.4xlarge. An additional 4-core instance, c3.xlarge, was used only for on-demand Squid servers and not for worker nodes

application required 2 GB or more. In addition, jobs could not run on any 4-core instance types because of the configuration of the FIFE factory, which submitted requests to HEP Cloud to provision slots only with 8-core and 15 GB of RAM. These parameters were hard to change because the factory was part of FIFE production operations. On the other hand, the 4-core instances were the most popular for CMS and were key in enabling the scale of 58,000 cores during the CMS demonstrator of HEP Cloud. In response, the design of HEP Cloud is being reconsidered to enable more flexibility in the resource allocation.

The full production campaign processed a dataset of about 57,000 files (56 TB). This was organized in multiple job submissions that processed the same draining dataset (§4). The 2nd resubmission consisted of 10,000 jobs, the maximum allowed by the FIFE Batch submission system, processing a maximum of 5 files per job from SAM. This job submission processed 46,858 files out of 57,000. With an average processing time per file of 5 hours (Figure 2), the expected duration of a job was about a day. The remaining files were processed by further recovery submissions, together with the files unprocessed because of application failure or preemption. Considering that AWS is a highly preemptive environment, this was considered an appropriate and acceptable operational practice.

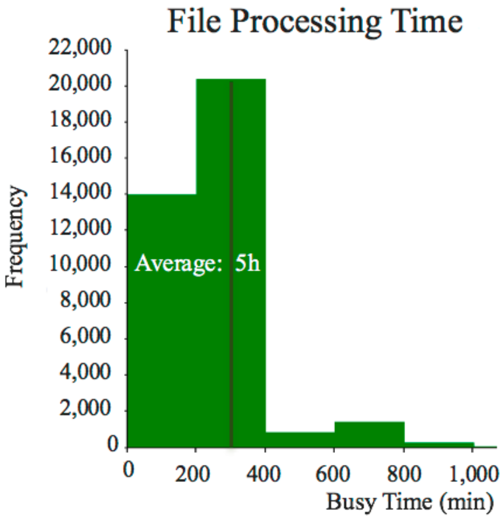


Figure 2: Distribution of the file processing time

To give a sense of scale on the amount of preemption that occurred during the run, Figure 3 shows the total amount of virtual machines running and preempted (“overbid”) every hour. In total, 1,035 virtual machines running up to 8 slots were preempted in 2 days. It should be noted that preempted jobs are automatically resubmitted by the system and do not necessarily result in a job failure. The file that was being processed at the time of preemption, however, was not presented by SAM again for processing and had to be recovered. As shown in Figure 4, only

37% of the jobs completed without being preempted, 38% of the jobs were resubmitted twice by the system, and so on.

Table 1 shows the efficiency of the workflow in terms of Wall time and CPU time. Considering only final jobs[8], the efficiency calculated as CPU time over Wall clock time varies from 73% to 96% depending on the resubmission. These are considered good efficiencies, considering that the workflow interacts with storage and external databases and, thus, is not CPU-bound. It should be noted that despite the large difference in wall clock times between all the jobs and the final jobs, in our processing scheme preempted jobs can contribute to processing files, as opposed to the CMS use case, for example. It would be a mistake, therefore, considering that the inefficiency due to preemption, generally calculated as the ratio of the two numbers, ranges to up to ~50%.

	Time	Subsequent Recovery
Sum of all jobs Wall Clock (h)	202,706	60,059
Sum of final jobs Wall Clock (h)	102,552	45,436
Sum of final jobs CPU Time(h)	74,958	43,589

Table 1: Wall clock and CPU hours for all the jobs and the final jobs

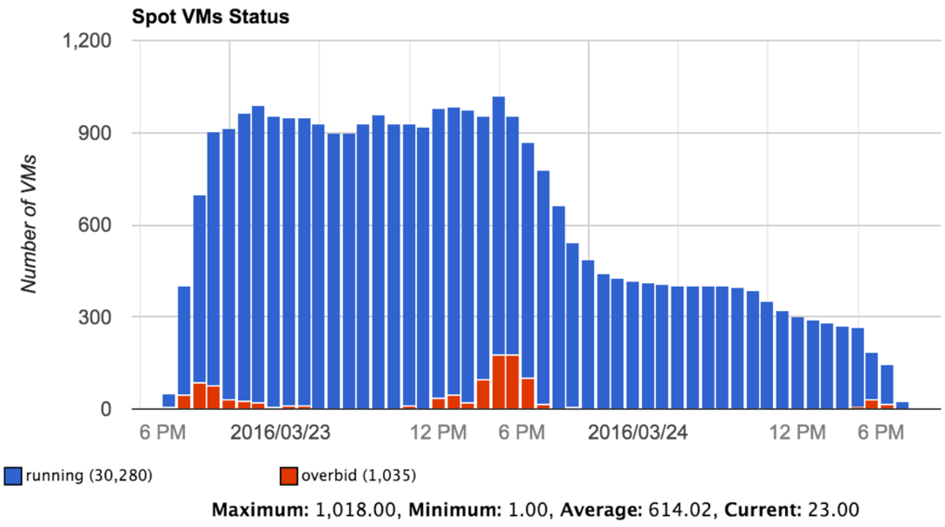


Figure 3: The total number of instances "running" and preempted ("overbid") every hour

[8] A limitation of HEP Cloud is that CPU time is available only for the final jobs.

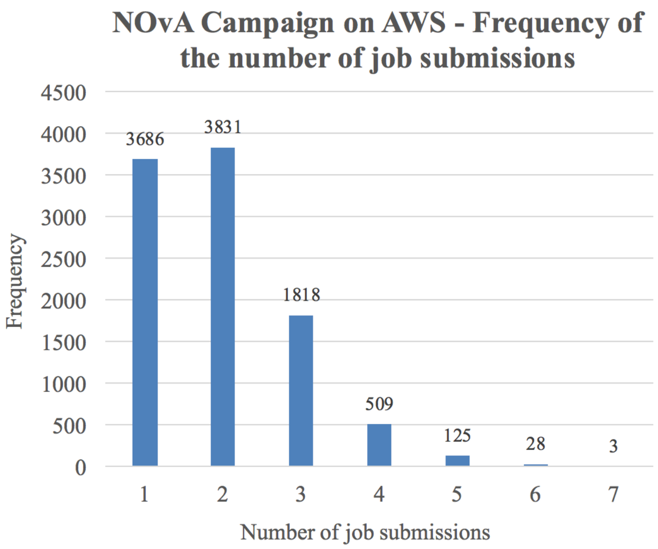


Figure 4: Frequency of the number of job submissions.

File Status	Files		File-based Efficiency		
	Number	Percentage	Total Time	Percentage	Subsequent Recovery
Consumed	31,076	66%	131,000	64%	78%
Failed	4,314	9%	3,400	2%	5%
Preempted	11,468	25%	69,000	34%	17%
Total	46,858	100%	203,400	100%	100%

Table 2: Number of files and file-based efficiencies for a given file status.

A good metrics to understand the effectiveness of the workflow in processing data is the file-based efficiency. It is calculated as the total wall clock time spent by the application in processing files of a given status over the total wall clock processing time (203,000 hours). Table 2 shows the status of the files at the end of the processing and the file-based efficiency. A "consumed" status results from a successful processing; "failed" results from application failure, typically problems accessing the calibration database (in this submission) or out of memory conditions; "preempted" indicates those files that were being processed when the VM was preempted. As seen in the table, despite the large number of job resubmission (63% as per Figure 4), the total number of failed and preempted files is 34%, pointing to the fact that many preemptions happened early in the campaign, when few files had started to be processed (Figure 3). Due to the network disconnection incident, the times for failed or preempted jobs tend to be higher than expected, leading to an efficiency of consumed file processing of 64%. The same metric for the subsequent recovery of the dataset, which had no notable incidents, is higher at 78%.

An important improvement for this submission was the full use of S3 for data staging. The dataset was prestaged to S3 and the input files were copied to the local scratch disk of the VM by the ifdh tool as directed by SAM. The Output was also stored in S3. Figure 5 shows that the achieved aggregated input bandwidth from S3 to the jobs was approximately 10 Gbps. This results from a maximum number of concurrent read accesses from the jobs around 800, with an input file size of 1 GB.

The standard SAM4Users tools were used to transfer the 124 TB of output from S3. AWS throttles data transfers from S3 to a single node on the internet at about 1 Gbps. The tools had to be enhanced to use multiple nodes to increase the bandwidth of data transfer.

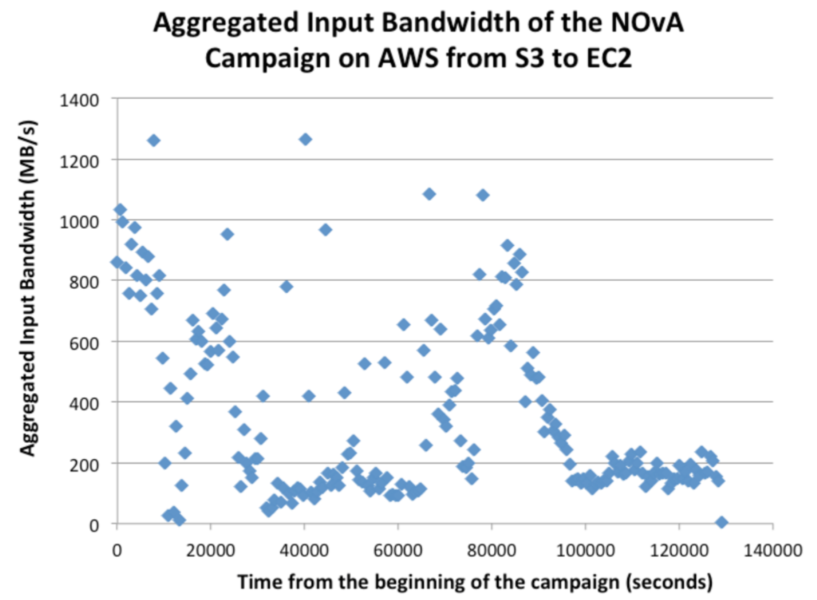


Figure 5: Aggregated input bandwidth from S3 to the jobs

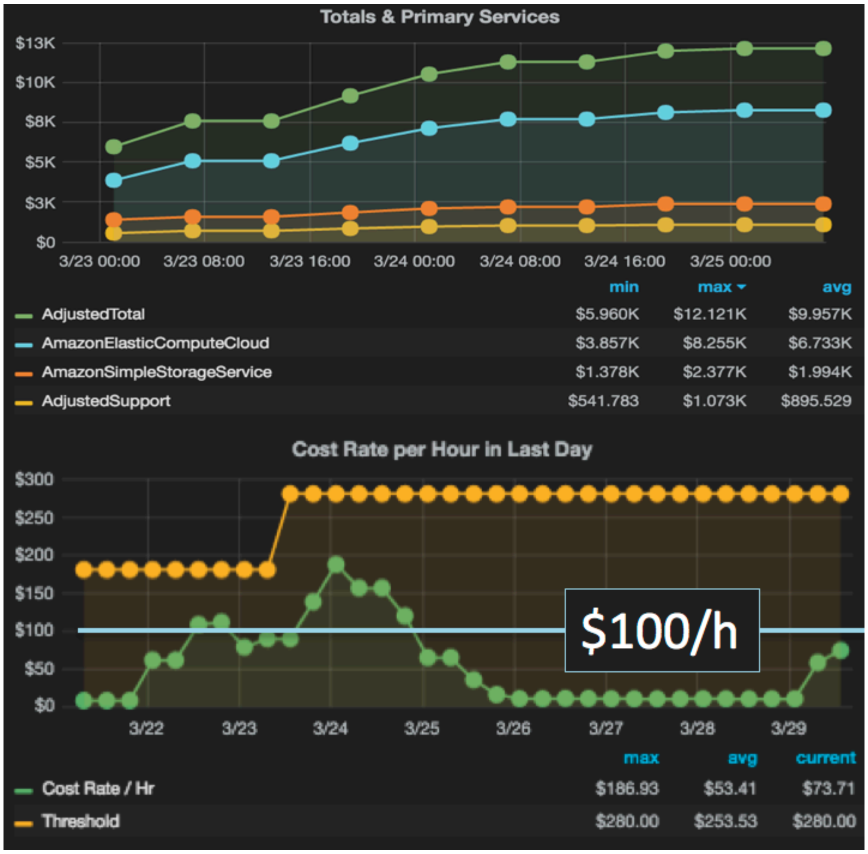
7 Costs

The cost of commercial clouds has steadily decreased in recent years and provisioning burst capacity on the cloud is becoming ever more attractive. Cloud costs include prices for the computation as well as data storage and movement. In the AWS model, the latter is free for data ingest but arguably expensive for egress. For scientific institutions, however, AWS contains the cost by waiving the fee of data egress if the monthly data movement cost is 15% or less of the total cost (data egress waiver: § 2). For the NOvA campaigns, in addition, AWS has waived all costs of data movement, even if above the 15% threshold, because their processes to enforce the waiver were still being refined. All data movement costs mentioned in this section are estimates.

To minimize costs, the data egress waiver billing scheme produces an “artificial” operational incentive to produce data and transfer it back within the calendar month boundary. In particular, for a single campaign, producing data one month and transferring it the next would not take advantage of the waiver. The model is complicated by the fact that the waiver is calculated over the aggregated costs of all Fermilab accounts (including CMS, NOvA, R&D, etc. today). Since we don’t envision that cross-experiment operational coordination is feasible in practice, individual experiments should only consider the operational properties of their workflow to estimate their costs.

The total cost of the 2nd submission of the near detector reconstruction amounted to $6,160 (Figure 6 top), broken down in

- $4,400 of EC2 costs, with $1,300 due to inter-region output transfers and the rest due to VM instance allocations.
- $1,000 of S3 costs, with $700 due to inter-region input transfers and the rest to storage
- $530 of AWS support



	min	max	avg
AdjustedTotal	$5.960K	$12.121K	$9.957K
AmazonElasticComputeCloud	$3.857K	$8.255K	$6.733K
AmazonSimpleStorageService	$1.378K	$2.377K	$1.994K
AdjustedSupport	$541.783	$1.073K	$895.529

	max	avg	current
Cost Rate / Hr	$186.93	$53.41	$73.71
Threshold	$280.00	$253.53	$280.00

Figure 6: Costs of running the NOvA campaign. Top: accruing costs for the campaign for computing (EC2), storage (S3), support and total. Bottom: the hourly cost rate averaged over the last day (green) and the spending alarm threshold used as security control (yellow).

As mentioned, this does not include any costs of data egress from S3 to Fermilab. The inter-region transfers, charged at $0.02 / GB, derived from storing the data at one AWS region and accessing / writing it from three. This cost was estimated to be less than replicating the entire 57 TB input and 124 TB output datasets to all three regions for an estimated month of processing at ~$0.03 / GB per month. In theory, by automating the process of data ingress and egress, the time needed for data storage at S3 could be further reduced, thus making replication more cost effective and, ultimately, reducing price.

When running at the scale of 7,300 slots, the cost rate was approximately $100 / h (Figure 6 bottom).

Due to the transparent cost structure, this experience has allowed us to evaluate the total cost of job failures, a number often difficult to calculate for standard Grid facilities. For this submission, the total cost of failure was $120. The relatively small amount was mainly due to the fast failure mode for most jobs. This cost does not include the jobs failed due to the network disconnection, since those were mostly preempted. The total cost of preempted jobs was $2,100, but it would be unfair to consider this a total loss since preempted jobs may contribute to processing files.

The Estimated total cost for running the full reconstruction campaign is $7,900 for 260,000 CPU h. The additional cost of data egress without the waiver is estimated at $9,400 for a total output dataset of 124 TB. The estimated ratio of the cost of data egress over the total cost is therefore about 54% for this workflow. The egress cost discounted by the 15% waiver would be $5,300. This calculation assumes that the computation and data egress can be done on the same calendar month, to take full advantage of the waiver. In general, workflows with a small data output as compared to their computation should be preferred as candidates for execution at AWS.

8 Cost Comparisons

The comparison of the costs to run at Fermilab and AWS are summarized in Table 3. The cost at Fermilab has been calculated including factors such as the amortization of the cost of the computing center building, power and cooling, cost of the hardware (computing, networking, etc.) and its lifetime, system administrators, etc. [1]

The cost for NOvA derives from the cost of support, computing, and storage, including inter-region transfers, but not the cost of data egress, as discussed above. The error is derived from the distribution of costs on a 6-hours basis.

Fermilab CMS Tier-1	$0.009 ± 25% per core-hour
CMS at AWS	$0.014 ± 12% per core-hour
NOvA at AWS	$0.029 ± 14% per core-hour
NOvA cost at AWS per consumed file	$0.20 per 2,000 events

Table 3: Cost comparisons between Fermilab and AWS for the NOvA and CMS campaigns

Although we don't have a direct comparison of worker node performance running the same NOvA workflows at Fermilab and AWS, we have executed benchmarks on both systems and the performance is very similar. In particular, we have executed a simulation of a $t\bar{t}$ production to mimic the behavior of a Monte Carlo workflow. The systems at Fermilab produced in average 0.0163 events / s per core while at AWS 0.0158 events / s per core; therefore, AWS was slightly slower. Benchmarks based on HEPSpec06 produced similarly comparable results. Given the similar performance, the cost comparison is based directly on the cost of core hours at Fermilab and at AWS.

As shown in the table, the cost of running at AWS was about three times larger than running locally. This premium is generally considered acceptable to achieve large burst capacity in case the additional slots could not be provisioned in any other way (e.g. on OSG or HPC in the future). The position of the experiment is that using commercial clouds shows its potential as a reliable way to meet peak data processing needs at times of high demands. This assumes that the cost of maintenance for the HEP Cloud facility will decrease to a low level as compared to the effort required to support this demonstrator.

Running the NOvA workflow at AWS was costlier than running the CMS one. This is because CMS was running a full chain event simulation, starting from the generation stage, requiring mainly a relatively small amount of pile-up data in S3, as opposed to NOvA that pre-staged all input in S3. CMS also streamed output directly to Fermilab, while NOvA stored data to S3 to enable fast job recovery. Therefore, NOvA incurred the additional costs of storage and inter-region transfers. Arguably, this additional cost was warranted in the NOvA processing model, since staging the output to S3 and immediately declaring the files to SAM simplified and accelerated recovery operations. In addition, NOvA's set up prevented it from accessing a variety of cheaper instances. This was in part because of the larger memory footprint of the application and in part because the setup of the FIFE submission system was harder to change, being in production, and 4-core instances could not be provisioned. These cheaper instances made 60% of the total number of instances in the CMS campaign.

9 Lessons Learned

Several HEP communities are exploring the model of using commercial clouds to complement local resources with burst capacity. Despite the transparency on the costs of cloud providers, the actual elements of scientific workflows that contribute to the total costs are not always evident. For example, CMS has learned that high I/O rates of data reads (GET) from S3 and remote reads over the WAN (high-latency) are costly in Amazon, thus full data transfers and local seeks are preferred. For NOvA, we have learned the cost structure in detail and the costs of high data output rate per unit of computation. In general, this experience has demonstrated that commercial clouds through HEP Cloud are a promising option to address peak needs.

On AWS, the large scale of affordable resources can be achieved only by a combination of instance types, availability zones and regions. To improve on the cost per core hour, there is an incentive in lowering the memory footprint of the workflow to gain access to more and sometimes cheaper instance types. This is in general true also for the OSG environment, although the cost implications are less evident. In addition, the design of HEP Cloud is being revised to enable greater flexibility in resource provisioning, rather than relying on external submission systems to define the number of cores and amount of memory for a computing slot.

For NOvA, the current scale of slots is limited by the FIFE Batch submission capacity. That is limited to 20,000 running slots for each of the two schedulers. This limit affects the overall capacity of the system, but it is particularly relevant when attempting resource bursts. In this scenario, operations also benefit from the ability to associate a large number of jobs with the same dataset processing context (SAM project). This reduces the job duration because it allows operations to submit more jobs for a given dataset and reduce the number of files processed per job. In turn, this lowers the risk of preemption and allows operations to keep bookkeeping simple avoiding the need of splitting the dataset in smaller subsets. These benefits are achieved today with single large submissions (up to 10,000 jobs) to allow operating on a single draining dataset. This could be achieved in the future by passing the same SAM project to multiple job submissions to go beyond the current job submission limit of 10,000.

To simplify operations with the NOvA (SAM) data processing model, the jobs stored data to S3 and declaring metadata and file location to SAM. This was key to enabling rapid turnaround of job recovery, since at the end of the jobs, the status of the processed files was up to date in the database. The explored alternative was sending the data back to Fermilab and allowing the File Transfer Service (FTS) to declare data asynchronously. The problem with this approach is that recovery had to wait a long time i.e. until the metadata was declared by FTS to SAM, to have an updated list of files to (re)process. The recovery logic could be automated further to simplify operations as much as possible. In general, this experience has forced us to evaluate the costs of data egress. Workflows that cannot take full advantage of the waiver should be scrutinized as to their suitability for Cloud computing.

When running jobs on the spot instance market of AWS, the efficiency based on the success rate of jobs tend to be lower than on local resources, due to the high occurrence of preemption. This encourages the perception of inefficient use of resources, while preempted jobs can contribute to file processing, despite not terminating successfully. We will work on a more automated calculation of efficiency based on file consumption status. This will reflect more closely the overall efficiency of data processing irrespectively of the preemption status of the jobs.

Finally, we found it extremely important to proactively transfer the lessons learned (e.g. configuration changes) from the context of one experiment to another.

10 Future Work

The HEP Cloud project has successfully concluded its pilot phase in June 2016. The new charge for a new phase of the project will focus on a five year plan. It will transition the use of Commercial Cloud resources to production operations in FY18. It will integrate High Performance Computers with the facility, with an eye towards DOE resources, starting from NERSC. It will automate provisioning based on the current allocation of resources and the capabilities and requests of the workflow; this will be implemented through the Decision Engine component. The project will explore the need and seek the Authority to Operate from DOE, by strengthening its security stance via comprehensive Risk Assessment and Security Plan documents, and operational documentation.

11 Conclusions

The HEP Cloud project has demonstrated the feasibility and cost effectiveness of using commercial clouds for burst capacity. The use cases explored targeted CMS, NOvA, and DES. CMS has demonstrated that they could achieve an additional 58,000 slots for 2 weeks, corresponding to a 25% increase in their total computing capacity. NOvA has executed simulation and reconstruction workflows on HEP Cloud. The scale achieved was 7,300 slots, equivalent to more than 3 times their local slot allocations at Fermilab. One of the workflows has been executed while CMS was running, demonstrating that HEP Cloud could manage both use cases at the same time. The cost was contained by using the AWS spot instance market, which reduces prices by almost an order of magnitude by accepting that the VM could be preempted with a two minutes' notice. This was not an operational burden for NOvA, since it used SAM, its familiar service for data handling. The jobs stored the files in the local storage at AWS (S3) and declared metadata and file locations to SAM. The system records the status of processed files and its parentage, allowing easy recovery of unprocessed data.

The file-based efficiency of the workflow execution the analyzed ranged from 64% to 78%. The largest workflow submission consisted of almost 47,000 files processed in two days with a plateau of 7,300 provisioned slots for a total of about 200,000 hours. Overall, the final campaign processed 57,000 input files for 56 TB of data and produced 124 TB of output. At the maximum rate, the cost was about $100/h without considering data egress. This resulted in a cost of $0.20 to process a file of 2,000 events. In terms of computations, the cost was $0.029 per core hour, which is a factor three higher than the estimated cost at Fermilab of $0.009 per core hour. In case of lack of available resources, this cost is generally considered acceptable by NOvA to satisfy their needs for burst capacity. In addition, the cost could be brought down by increasing the flexibility of resource provisioning by HEP Cloud, rather than relying on external submission systems to define the resources for a computing slot, and by reducing the memory footprint of the NOvA workflow to get access to cheaper VM instances. With these cheaper instances, the average cost of the CMS campaign was $0.014 per core hour.

As the HEP Cloud project goes into production in FY18, Cloud computing has the potential to become a cost effective and user friendly solution for resource bursts.

12 References

1] L. Bauerdick, B. Bockelman, D. Dykstra, I. Fisk, S. Fuess, G. Garzoglio, M. Girone, O. Gutsche, B. Holzman, H. Kim, R. Kennedy, D. Hufnagel, N. Magini, D. Mason, P. Spentzouris, A. Tiradani, S. Timm and E. Vaandering, "HEPCloud, a new paradigm for HEP facilities: CMS Amazon Web Services Investigation," FERMILAB-PUB-16-170-CD, 2016.

2] S. Timm, G. Garzoglio, P. Mhashilkar, J. Boyd, G. Bernabeu, N. Sharma, N. Peregonow, H. Kim, S. Noh, S. Palur and I. Raicu, "Cloud Services for Fermilab Stakeholders," *J. Phys.: Conf. Ser. 664 022039,* 2015.

3] A. W. Services, "AWS Offers Data Egress Discount to Researchers," 2016. [Online]. Available: https://aws.amazon.com/blogs/publicsector/aws-offers-data-egress-discount-to-researchers/. [Accessed 17 June 2016].